

05 初心 6 問



Created by GT F

Last updated 2019-08-11

- {} の意味
- インデント(intent)は大切。
- インデントと変数スコープ
- 設問
- ヒント

{ } の意味

以下質問を回答してください。1. "aaa" の出力回数? 2. "bbb" の出力回数?

```
1  for (int i = 0; i < 2; ++i) {  
2      for (int j = 0; j < 3; ++j) {  
3          System.out.println("aaa");  
4      }  
5      System.out.println("bbb");  
6  }
```

インデント(intent)は大切。

プログラムを理解しやすいため、ほとんどのプログラムはインデント（文字下）を利用します。Java のプログラミンの一般的の規則は、新しいブロック文や条件文が始まる度々に 1 つインデントを追加します。インデントの文字数はプログラマを決定しますが、一般的Javaは 4 文字の空白（スペース）を利用します。

インデントとコードスタイルの一例

```
13  public class IntentSample {  
14        
15      public static void main(String... args) {  
16          int x = 100;  
17          for (int i = 0; i < x; ++i) {  
18              System.out.println(x);  
19            
20          if (i % 2 == 0) {
```

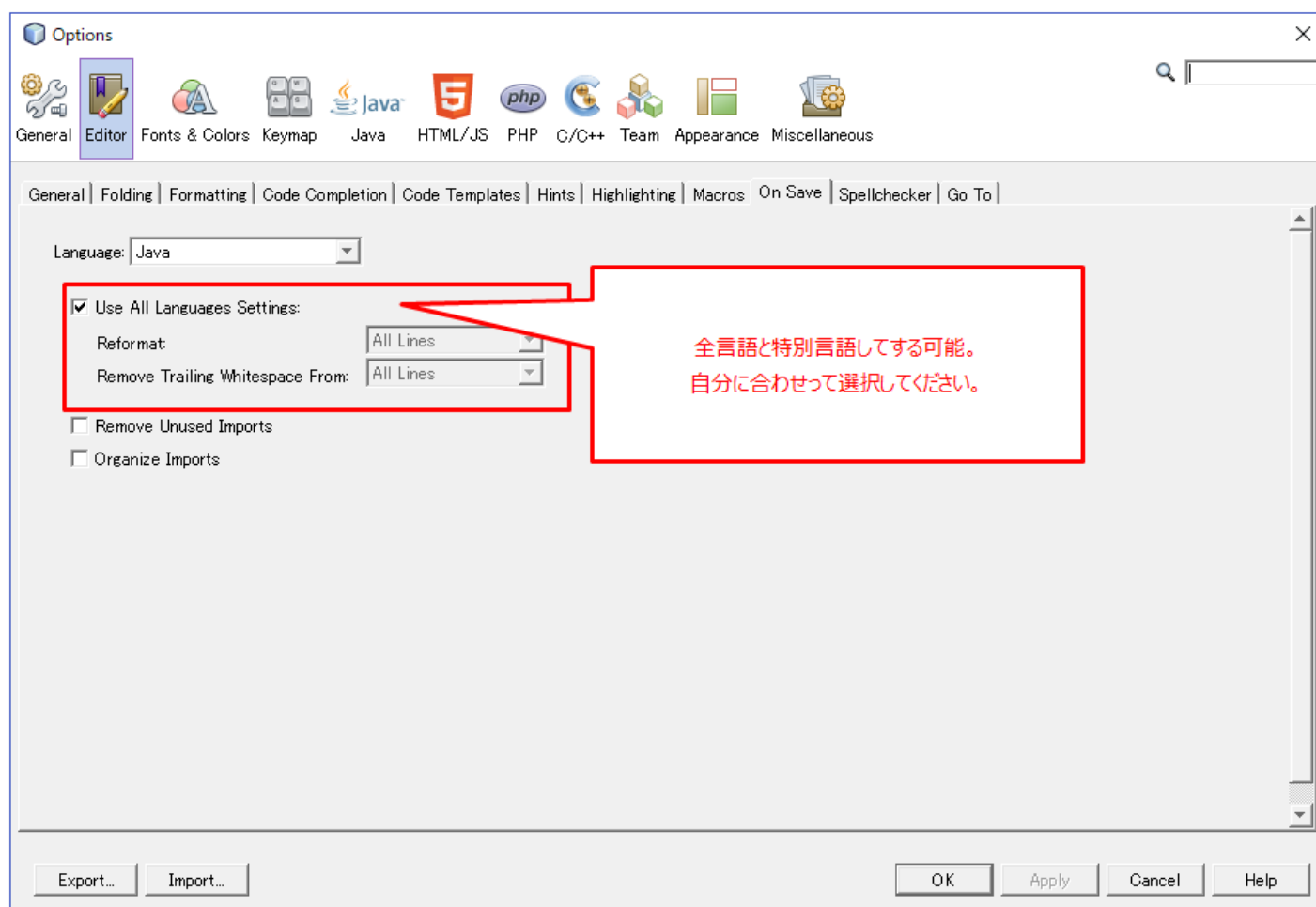
```

21 ..... System.out.println(i);
22 ..... }
23 ..... }
24 ..... }
25
26 }

```

自動的にインデントOR自動的にコードフォーマットする可能。Netbeans(シヨットカットのキーセ
ットはEclipseマッピングする)は、以下手順通り、自動的にコード整形設定可能。

メニューバー / ツール(tool) / オプション(option) / エディター(editor) / 保存の時 (onSave)



⚠ コードフォーマットは自動整形機能を依頼ずに、プログラマの基本能力であること。

インデントと変数スコープ

インデントは変数が見えるか見えないかを示します。

```

13 public class IntentSample {

```

```

14
15 public static void main(String... args) {
16     int x = 100;
17     for (int i = 0; i < x; ++i) {
18         System.out.println(x);
19
20         if (i % 2 == 0) {
21             for (int z = 0; z < 20; ++z) {
22                 System.out.println(z);
23             }
24         }
25     }
26 }
27
28
29 }

```

変数 x が見える

変数 x, i が見える

変数 x, i, z が見える

設問

- 設問 1 : 1から100までの偶数の和を求める(for + if)。
- 設問 2 : 5の階乗を求める(for)。
- 質問 3 : 配列 {1, 2, 6, 7, 9, 6, 2, 1} の最大値、最小値、和、平均値を求める。
- 質問 4 : 質問 3 の配列に重複ある要素を探す (要素の表示回数集計必要なし) 。
- 質問 5 : 質問 3 の配列に値「7」のインデックスを求む。
- 質問 6 : 任意行列 (2次元配列) の積を求める。
- 質問 7 : 配列を昇順でソートする (java.util等メソッド利用禁止) 。
- 質問 8 : LV10のパスカルの三角形を出力してください。

ヒント

例 : 設問 1 : 1から100までの偶数の和を求める(for + if)。

STEP1 : 1から100まで ⇒ 繰り返す。繰り返す抜き出す条件は100以上超える。

```

1 for(int i = 1; i < 101; i++) {
2     // 処理追加
3 }

```

STEP2 : 和を求める

```

1 int sum = 0;
2 for(int i = 1; i < 101; i++) {

```

```
3    sum = sum + i; // 1~100の和を求める
4 }
```

STEP3 : 偶数のみの値⇒（偶数の加算するロジックを追加必要）⇒偶数を判断するロジック（if）

```
1  if(i % 2 == 0) {
2      // 偶数が判断される
3  }
```

STEP4 : 上記1,2,3をマージして、プログラミング完成！

例：設問3：配列 {1, 2, 6, 7, 9, 6, 2, 1} の最大値、最小値、和、平均値を求める。

STEP1 : 配列 {1, 2, 6, 7, 9, 6, 2, 1} の宣言。

```
1  int[] array = new int[] {1, 2, 6, 7, 9, 6, 2, 1};
```

STEP2 : 配列の各要素をアクセスする

```
1  for(int i = 0; i < array.length; i++) {
2      System.out.println("現在のインデックスは " + i);
3      System.out.println(array[i]); // 配列の各要素を出力する。
4  }
```

STEP3 : 最大値を求める為に、分岐が必要

```
1  int[] array = new int[] {1, 2, 6, 7, 9, 6, 2, 1};
2  int max = 0; // 最大値を一時に値を設定する。MAXの初期値は0を設定していいですかを考えてください。
3  for(int i = 0; i < array.length; i++) {
4      if(array[i] > max) { // array[i]配列の現要素 を 前回取得した最大値 比較する
5          max = array[i];
6      }
7  }
8  System.out.println("配列に最大値=" + max); // 配列に最大値=9
```

 Like Be the first to like this

No labels 